המאמר היומי של מייק - 07.01.25  
Anchored Preference Optimization and Contrastive Revisions Addressing Underspecification in Alignment

המאמר שנסקור היום מציע שיפור לשיטת יישור (alignment) למודלי שפה, DPO, השייכת למשפחת טכניקות RLHF או Reinforcement Learning with Human Feedback. כמו שאתם זוכרים RLHF הינו אחד השלבים(האחרון בד״כ) לאימון LLM יחד עם אימון מקדים (pretraining) ו-Supervised Fine Tuning או SFT בקצרה.

מטרת RLHF היא להראות למודל מה ההבדל בין תשובות מועדפות (על ידי בני אדם) מתשובות פחות מועדפות. בנימה יותר מתמטית RLHF מאמנת את המודל למקסם את היחס בין הציון של התשובה מועדפת (טובה) יותר לבין תשובה פחות טובה. שיטת RLHF קלאסית Proximal Policy Optimization מוסיפה לאיבר הממקסם פונקציית לוס איבר רגולריזציה המנסה לשמור את הפוליסי הנלמד (כמו LLM מאומן) קרוב ל-LLM ההתחלתי (הקרבה מחושבת עם KL על ההתפלגות של הטוקנים החזויים על ידי שני המודלים).

הציון מחושב על מודל תגמול (reward model) שמאומן (בלשב הקודם ל-RLHF) לשערך את ״איכות״ התשובה לשאלה נתונה. כלומר מודל תגמול R אמור לתת ציון גבוה לתשובה טובה וציון נמוך לתשובה פחות טובה. המודל מאומן על זוגות של תשובות טובות ולא טובות לשאלות, כאשר בד״כ התיוג של התשובות מתבצע על ידי מתייגים אנושיים (לפעמים רותמים מודל שפה עוצמתי לתיוג הזה).

התברר שניתן לקרב את יעד האופטימיזציה של PPO ללא אימון של מודל תגמול. בשנתיים האחרונות יצאו לא מעט מאמרים שהציעו שיטות ש״יודעות״ להסתדר ללא מודל תגמול. אחת מהן היא DPO שזה ראשי תיבות של Direct Preference Optimization. עם DPO פונקצית תגמול מוגדרת r\_dpo בתור לוגריתם של היחס בין הפוליסי (ההתפלגות החזוי של טוקנים הנמדדת על ידי המודל או נראות- likelihoods) עבור המודל המאופטם (שעובר פיין טיון) לבין זה של המודל ההתחלתי. מטרת אימון DPO היא למקסם את הפרש בין התוחלת (עבור הדאטהסט של זוגות שאלות ותשובות) ההפרש של r\_dpo ביו התשובות המועדפות לבין פחות מועדפות.

הנקודה העיקרית של המאמר היא האובזרבציה שהאופטימיזציה של פונקצית המטרה של DPO עלולה להשפיע באופנים שונים על יחס הנראויות (likelihoods) של תשובות המועדפות w לאלו של פחות מועדפות l. היא כמובן יכול להגדיל את ההפרש ביניהם (שזה המטרה המוצהרת שלה) אבל יכול להגדיל את p\_w יותר מאשר הוא מגדיל p\_l, או להקטין את p\_l יותר מאשר הוא מקטין את r\_w. תרחישים אלה עשויים להוביל ליצירת מודלים שונים מאוד. המאמר מציין שתשובה מועדפת אינה בהכרח טובה יותר ממה שהמודל מייצר לפני היישור. במקרה זה, DPO עלול לפגוע בביצועי המודל.

המאמר מתבונן במקרים השונים של ערכי r\_dpo עבור התשובות w ו- l(מועדפת ופחות מועדפת בהתאמה) ובונה שתי פונקציות מטרה ל- DPO שעשויות להוביל לביצועים טובים יותר עבור מקרים אלו. שיטת אימון שמאפטמת פונקציות אלו קיבלה שם Anchored Preference Optimization או APO. הפונקציה המוצעת הראשונה מגדילה את ערך הפוליסי (נראות של תשובה) כאשר הערך הנוכחי של r\_dpo עבור w קרוב ל-0 (w הינה בעלת נראות נמוכה יותר עבור המודל ההתחלתי) ומקטינה את הנראות של התשובה הפחות מועדפת עוד יותר אם r\_dpo עבור l קרובה ל-0.

הפונקציה המוצעות השניה לעומת זאת מקטינה את הנראות של w כאשר r\_dpo קרוב ל -0 עבור w ומגדילה את ההפרש בין הנראויות של w ו- l כאשר ההפרש בין r\_dpo עבור w ו- l קרוב ל-0. כל זה במטרה לגרום למודל שפה המאומן באמצעות DPO להתכנס לפתרון טוב יותר.

יש עוד משהו מעניין במאמר הזה. המחברים טוענים שכדי ש- DPO יעבוד בצורה טובה יותר, שתי התשובות(w ו- l) צריכות להיות רלוונטיות לשאלה ואחת מהן צריכה להיות ״רק קצת״ יותר טובה מהשנייה. כלומר במו בלמידה ניגודות עדיף לאמן את המודל על hard negatives.

המחברים מציעים שיטה לזיהוי (ובניית דאטהסט) של תשובות מועדפות ופחות מועדפות והיא יצירת תשובה מועדפת מתשובה כלשהי(אך רלוונטית) על ידי הפעלת LLM המשפר את התשובה (עם פרומפט מתאים). שיטה אחרת שהמחברים מציעים להשתמש בה היא בהינתן שתי תשובות של המודל המאומן (עם DPO) להפעיל מודל שפה שמטרתו להגיד מהי תשובה טובה יותר (זה נקרא on-policy judge). ניתן גם לבנות דאטהסט באופליין עם מודל שפה שלישי ומודל שופט.

סקירה ארוכה - אני מקווה ששרדתם…

https://arxiv.org/abs/2408.06266